Proc. of the 7th Annual International Conference on Mobile Computing and Networking (MOBICOM 2001), Rome, Italy, July 2001.

The Cricket Compass for Context-Aware Mobile Applications

Nissanka Priyantha, Allen Miu, Hari Balakrishnan and Seth Teller

MIT Laboratory for Computer Science
{bodhi, aklmiu, hari, seth}@lcs.mit.edu

Abstract. The ability to determine the orientation of a device is of fundamental importance in context-aware and location-dependent mobile computing. By analogy to a traditional compass, knowledge of orientation through the *Cricket compass* attached to a mobile device enhances various applications, including efficient way-finding and navigation, directional service discovery, and "augmented-reality" displays. Our compass infrastructure enhances the spatial inference capability of the Cricket indoor location system [20], and enables new pervasive computing applications.

Using fixed active beacons and carefully placed passive ultrasonic sensors, we show how to estimate the orientation of a mobile device to within a few degrees, using precise, sub-centimeter differences in distance estimates from a beacon to each sensor on the compass. Then, given a set of fixed, active position beacons whose locations are known, we describe an algorithm that combines several carrier arrival times to produce a robust estimate of the rigid orientation of the mobile compass.

The hardware of the Cricket compass is small enough to be integrated with a handheld mobile device. It includes five passive ultrasonic receivers, each 0.8cm in diameter, arrayed in a "V" shape a few centimeters across. Cricket beacons deployed throughout a building broadcast coupled 418MHz RF packet data and a 40KHz ultrasound carrier, which are processed by the compass software to obtain differential distance and position estimates. Our experimental results show that our prototype implementation can determine compass orientation to within 3 degrees when the true angle lies between ± 30 degrees, and to within 5 degrees when the true angle lies between ± 40 degrees, with respect to a fixed beacon.

1 Introduction

Context-aware applications, which adapt their behavior to environmental context such as physical location, are an important class of applications in emerging pervasive computing environments [17]. Examples include *location-aware* applications that enable users to discover resources in their physical proximity [14, 20], active maps that automatically change as a user moves [22], and applications whose user interfaces adapt to the user's location. A significant amount of previous work has focused on providing device position capability indoors, including the Active Badge [26], Bat [14], RADAR [5], and Cricket [20] systems.

An important aspect of context, which is related to physical position, is the *orientation* of a device (or user) with respect to one or more landmarks in a region. A pervasive computing application can benefit from knowing this information, for instance by providing the ability to adapt a user interface to the direction in which a user is standing or pointing. Our first motivating application is called the *Wayfinder*. We envision this application to run on a handheld computer and help sighted or blind people navigate toward a destination in an unfamiliar setting. For example, the Wayfinder might lead a visitor from the entry lobby of a building to the office of the person hosting the visitor, or to a seminar room. The Wayfinder gives incremental directions to the user on dynamically retrieved ("active") maps [22, 20], using the user's position and orientation with respect to a fixed set of wireless beacons placed throughout the building. The second motivating application is called the *Viewfinder*. The user can point it in any direction, and specify a "sweep angle" and maximum distance. Using an active map integrated with a resource discovery system (e.g., the Intentional Naming System, INS [1]), the Viewfinder then retrieves and displays a representation of the set of devices and services lying inside the sector of interest specified by the user and allows the user to interact with these services via the representation on the map. A third motivating application is in the design of "augmented-reality" displays, where the user's view of the environment is overlaid with information about other objects present within that environment, and adapts to the direction that the user is looking toward [4, 25].

The underlying capability required for these applications is akin to a "software compass," which, endowed with a semantic map of its context and accurate knowledge of its own position and orientation, can inform the user of interesting resources and how to get to those resources. This paper describes the design and implementation of the *Cricket compass system*, consisting of a set of active beacons, passive hardware sensors,

and associated software algorithms. This system enables a robust software compass capability for a handheld device moving about inside a building.

The operating environment in the Cricket architecture is instrumented with active beacons, each of which broadcasts its own known position over an RF channel together with an ultrasonic pulse [20]. One RF receiver and several passive ultrasonic position receivers are precisely placed on a compass board. Software running on-board uses the *differentials* in distances reported by the ultrasonic receivers to infer the orientation (or "heading") of the device. The Cricket compass reports position and orientation indoors for a handheld, mobile device, and informs an application running on the device of the position and orientation in a local coordinate system established by the fixed set of beacons.

The first challenge in deriving *orientation* for a small device arises from the need for very accurate differential distance estimates: estimating orientation to within a few degrees of the correct value requires differential distance estimates to be of sub-centimeter accuracy, which is at least an order of magnitude smaller than the currently best available linear distance estimation technologies. We show how to do this using multiple carefully placed receivers. The second challenge arises due to variation in the speed of sound due to temperature and humidity, which affects the accuracy of position estimates. Rather than explicitly measuring this parameter with climate sensors, we calculate it directly from observed propagation times.

The Cricket compass system proposed in this paper addresses several problems with existing methods for orientation estimation. A traditional magnetic compass can estimate orientation, but exhibits enormous errors when near magnetic or time-varying electric fields, both of which are rather common in most modern buildings with computers and other equipment. Orientation can be inferred from a moving position sensor, but this usually requires large or fast user motions, which is undesirable in several applications. Active sensors on user devices typically lead to systems that track users [14, 26], which suffer from potential compromises to user privacy [19, 20]. In contrast, Cricket requires a small number of beacons at known positions in each room to instrument a building, but enables location and orientation for a passive handheld device without requiring any user motion. We have built several prototype beacons and a receiver compass configuration, and report experimental data that show that our software compass correctly estimates orientation to within a few degrees. We also describe a Viewfinder application developed using this capability. The Cricket

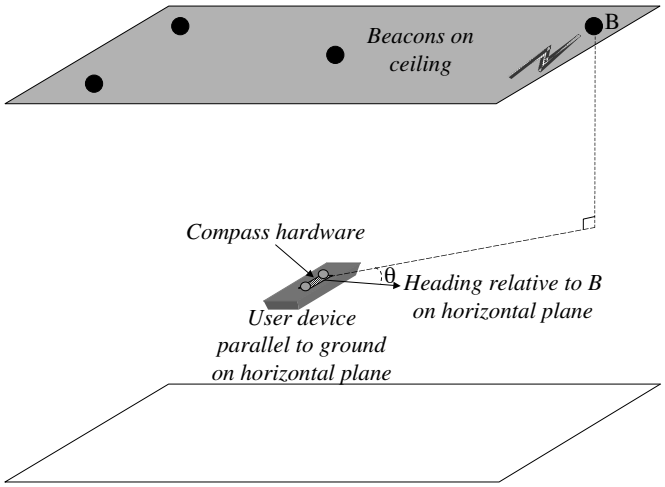

Fig. 1. Setup of beacons on the ceiling of a room and a user device with attached compass hardware. The Cricket compass solves the problem of obtaining the precise position and orientation of the user device relative to a coordinate system defined by the beacons.

system is being used in MIT's Project Oxygen in a variety of pervasive computing scenarios [17].

The rest of this paper is organized as follows. Section 2 details the design of the Cricket compass, describing its theory of operation, differential distance estimation, and coordinate determination algorithms. Section 3 discusses our implementation and Section 4 presents experimental results and an error analysis. Section 5 discusses some improvements based on our experimental results. Section 6 describes the Viewfinder application. We compare our system to previous work in Section 7 and conclude in Section 8.

2 Design of the Cricket Compass

Figure 1 shows a user device with attached compass hardware in a room with beacons placed on the ceiling. When the device is held parallel to the horizontal plane, θ is the angle formed by the heading direction shown, with the point where the perpendicular from beacon B inter-

sects the horizontal plane. We are interested in precisely estimating θ.

The basic idea is to use one RF receiver to receive coordinate information from the beacons, and multiple, carefully placed, ultrasonic receivers on the compass attached to the device to obtain the *differential* distance estimates of a beacon to each ultrasonic receiver. θ is a function of the differential distance of the linear distance of the compass from the beacon, and of the height of the beacon (ceiling) above the plane of the compass. We obtain per-beacon linear distance estimates by differencing the arrival times of coupled RF and ultrasonic signals sent from each beacon [20]. To obtain the height of the beacon from the compass, we estimate the position coordinates of the compass from the position coordinates disseminated by multiple nearby beacons.

The rest of this section describes how this idea can be realized in practice. We start by describing how directional information can be obtained using differences in distance between a beacon and different receivers. We describe a technique to achieve the required precision of differential distance estimates, using differential phase information of the ultrasonic waves reaching the receivers. Finally, we show how to obtain accurate position coordinate information without explicit knowledge of the speed of sound, compensating for its variation with physical conditions.

2.1 Theory of operation

Figure 2 shows a beacon B, and a compass with two ultrasonic receivers, R_1 and R_2, which are located at a distance L apart from each other. The angle of rotation of the compass, θ, with respect to the beacon B, is related to the difference in distances d_1 and d_2, where d_1 and d_2 are the distances of receivers R_1 and R_2 from B. The vertical and horizontal distances from the center of the compass to B are denoted by z and x, respectively.

Figure 3 shows the beacon B from Figure 2 projected on to the horizontal plane along which the compass is aligned. In this figure, x_1 and x_2 are the projections of distances d_1 and d_2 on to the horizontal plane. We assume that the compass is held parallel to the horizontal plane.

From Figure 2:

$$x_1^2 = d_1^2 - z^2 \tag{1}$$

$$x_2^2 = d_2^2 - z^2 \tag{2}$$

$$x = \sqrt{\bar{d}^2 - z^2}$$

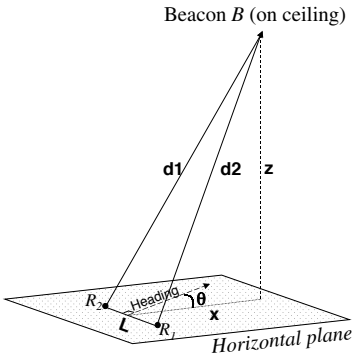

Fig. 2. Determining the angle of orientation along the horizontal plane, θ, using distance estimates. Observe that the heading is perpendicular to the line joining the ultrasonic compass receivers, R_1 and R_2, which are placed at a distance L from each other.

where $\bar{d} \approx \frac{d_1+d_2}{2}$ when $d_1, d_2 \gg L$.

From Figure 3:

$$x_1^2 = (\frac{L}{2}\cos\theta)^2 + (x - \frac{L}{2}\sin\theta)^2$$

and

$$x_2^2 = (\frac{L}{2}\cos\theta)^2 + (x + \frac{L}{2}\sin\theta)^2$$

$$\Rightarrow x_2^2 - x_1^2 = 2Lx\sin\theta$$

Substituting for x_1^2 and x_2^2 from Equations (1) and (2), we get:

$$\sin\theta = \frac{d_2 + d_1}{2Lx} \cdot (d_2 - d_1) \tag{3}$$

This may be rewritten as:

$$\sin\theta = \frac{d_2 - d_1}{L\sqrt{1 - (\frac{z}{d})^2}} \tag{4}$$

Equation (4) implies that it suffices to estimate two quantities in order to determine the orientation of the compass with respect to a beacon: (i) $(d_2 - d_1)$, the difference in distances of the two receivers

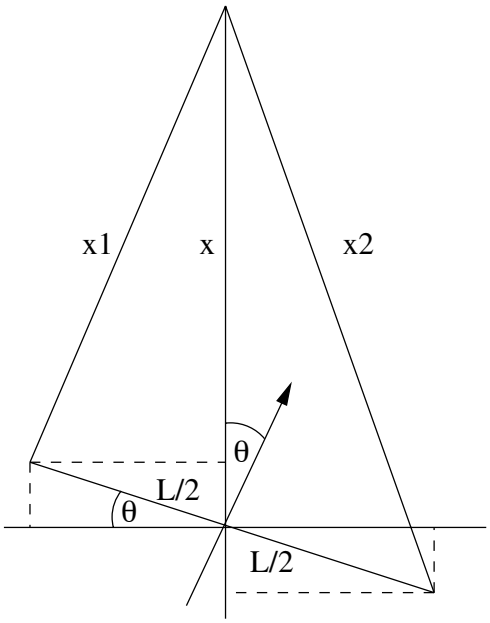

Fig. 3. A rotated compass leads to a difference in distances between the beacon and each of the receivers. This figure is the result of projecting the beacon onto the horizontal plane of the compass.

from the beacon, and (ii) $z/\bar{d}$, the ratio of the height of the beacon from the horizontal plane on which the compass is placed to the distance of the beacon from the center of the compass. In practice, however, no measurements are perfect. Our goal is to estimate each of these quantities with high precision, so as to produce a sufficiently accurate estimate of θ.

One way of precisely estimating $(d_2 - d_1)$ would be to precisely measure d_1 and d_2 separately, but that is easier said than done. Consider, for example, a situation where $L = 5$cm, and $\theta = 10^\circ$, with a beacon at a distance of 2 meters and a height of 1 meter from the receivers. From Equation (4), the value of $(d_2 - d_1)$ in this case is only ≈ 0.6cm, which is about an order of magnitude smaller than what current technologies can achieve in terms of linear distance estimates [14, 20][1]. Since our goal is to devise a compass with physically small dimensions, comparable in

[1] The worst-case error in $(d_2 - d_1)$ is equal to the sum of the errors in d_1 and d_2.

size to handheld PDAs, and still achieve high directional accuracy, we need an alternative method to estimate this differential distance.

Our solution to this problem tracks the *phase difference* between the ultrasonic signals at two different receivers and processes this information. We find that this approach allows us to obtain differential distance estimates with sub-centimeter accuracy. This is described in Section 2.2.

The second quantity, $z/\bar{d}$, is estimated by determining the (x, y, z) coordinates of the compass with respect to the plane formed by the beacons (the xy plane). We do this by placing multiple beacons in a room and estimating the time it takes for the ultrasonic signal to propagate between them and the compass. However, because the speed of sound varies with ambient temperature and humidity, we must estimate this quantity as well. This is described in Section 2.3.

2.2 Estimating differential distance

Consider two ultrasonic receivers R_1 and R_2 located a distance L apart, as shown in Figure 4. Let d_1 and d_2 be the distances to receivers R_1 and R_2 from beacon B. Let $\delta d = d_1 - d_2$ and let W_1 and W_2 be the ultrasonic waveforms received by R_1 and R_2 from B. The phase difference between the waveforms at the two receivers, ϕ, depends on the difference in distances traversed from B to the receivers by the ultrasonic signal and the wavelength λ of the signal, and may be expressed as:

$$\phi = \frac{(\delta d)}{\lambda} \cdot 2\pi \tag{5}$$

We call this the *actual* phase difference between the two signals and denote it by ϕ.

Because it is difficult to correctly determine the start of periodic waveforms, we can only obtain estimates for a waveform's phase in the range $(-\pi, \pi)$ from repeated low-to-high transitions of the signal. Unfortunately, a given observed phase difference between two waveforms, α, can correspond to an infinite number of actual phase differences, all separated by 2π. This in turn leads to multiple possibilities for δd.

One way to solve this problem is to observe from Equation (5) that as long as $\delta d < \lambda/2$, $\phi = \alpha$, and there is no ambiguity. Since d_1, d_2, and L are three sides of a triangle, $L \geq |d_1 - d_2| = |\delta d|$, and we can therefore place the receivers at a distance $L < \lambda/2$ to unambiguously determine ϕ and therefore uniquely estimate $(d_1 - d_2)$.

For a 40 KHz ultrasonic waveform at a temperature of 25°C and 50% humidity, $\lambda/2 = 4.35$ mm. This is smaller than the size of most

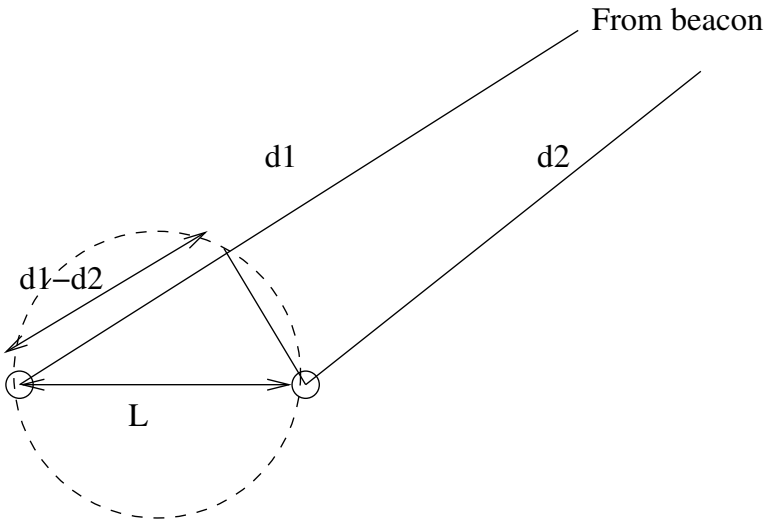

Fig. 4. Receivers R_1 and R_2 can measure the differential distance from a far-away beacon.

available ultrasonic signal receivers, which are typically on the order of about 1 cm. Lowering the carrier frequency is not an option, since this would make it audible to humans. We therefore need a nice general method to place receivers to unambiguously determine the phase difference.

One way of tackling this is to carefully place *three* receivers along a line, as shown in Figure 5, and use a *pair* of observed phase differences to estimate an actual difference. The intuition is that if the two inter-receiver distances, L_{12} and L_{23} are chosen carefully, then the actual phase difference between receivers 1 and 2 (say) can be disambiguated by using the phase difference between receivers 2 and 3, since the two phase differences are not independent.

Let ϕ_{12} and ϕ_{23} be the actual phase differences of a beacon's waveform between receivers 1 and 2 and receivers 2 and 3, respectively. Then,

$$\phi_{ij} = 2n_{ij}\pi + \alpha_{ij}$$

for each pair of receivers (i, j), where n_{ij} are integers and $-\pi < \alpha_{ij} \leq \pi$. Because the actual phase difference between two receivers is proportional to the distance traversed by the signal from the beacon to each of the receivers, $\phi_{23}/\phi_{12} = (d_2 - d_3)/(d_1 - d_2) \approx L_{23}/L_{12}$ when $d_i \gg L_{ij}$. This is shown in Figure 5.

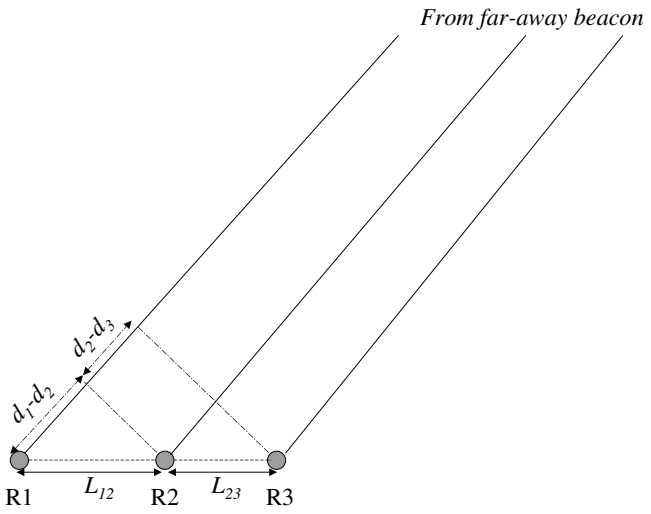

Fig. 5. Using three receivers to measure $(d_1 - d_2)$.

What we will show is that it is possible to pick L_{12} and L_{23} such that one can use two sets of observed phase differences α_{12}, α_{23} to *unambiguously* estimate the actual phase difference ϕ_{12}. In particular, we show the following result:
If L_{12} and L_{23} are relatively prime multiples of $\lambda/2$, then it is possible to use α_{12} and α_{23} to unambiguously obtain the actual phase differences ϕ_{12} and ϕ_{23}.

We argue this by contradiction. Suppose in fact there are two possible actual phase differences corresponding to a given observed phase difference for each receiver. For pair (i, j), call these differences ϕ'_{ij} and ϕ''_{ij}. Then, the following sets of equations hold:

$$\phi'_{ij} = 2n'_{ij}\pi + \alpha_{ij}$$

$$\phi''_{ij} = 2n''_{ij}\pi + \alpha_{ij}$$

Since each observed ϕ_{12} is related to the corresponding ϕ_{23} by the ratio L_{23}/L_{12}, the above equations can be rewritten as:

$$2n'_{23}\pi + \alpha_{23} = (L_{23}/L_{12})(2n'_{12} + \alpha_{12}) \tag{6}$$

$$2n''_{23}\pi + \alpha_{23} = (L_{23}/L_{12})(2n''_{12} + \alpha_{12}) \tag{7}$$

Subtracting Equation (7) from Equation (6) and rearranging, we get:

$$L_{12}(n'_{23} - n''_{23}) = L_{23}(n'_{12} - n''_{12}) \tag{8}$$

Let us express L_{ij} as $l_{ij}\lambda/2$, which expresses the separation between receivers as an integral multiple of $\lambda/2$. Equation (8) is then equivalent to:

$$l_{12}(n'_{23} - n''_{23}) = l_{23}(n'_{12} - n''_{12}) \tag{9}$$

where each of the l_{ij} and n_{ij} are integers.

Notice that $|n_{ij}|\lambda \leq \delta d$, the separation in distance between the carrier waveforms at receiver i and receiver j, and $\delta d \leq L_{ij} = l_{ij}\lambda/2$, for each pair $(i, j) = (1, 2), (2, 3)$. This means that $|(n'_{ij} - n''_{ij})\lambda| < 2L_{ij} = l_{ij}\lambda$. (In fact, $|(n'_{ij} - n''_{ij})\lambda|$ may be *equal to* $2L_{ij}$, but only if the beacon lies on the same horizontal plane as the compass. This situation is unlikely in practice, and detectable if it does occur.) Therefore, $|n'_{ij} - n''_{ij}| < l_{ij}$. Thus, if Equation (9) is to be satisfied, l_{12} and l_{23} *cannot* be relatively prime.

Hence, it is possible to unambiguously derive an actual phase difference (ϕ_{ij}) in the range of $[0, L_{ij}]$ from an observed one (α_{ij}) by picking L_{12} and L_{23} to be relatively prime integral multiples of $\lambda/2$. For example, we can pick $L_{12} = 2\lambda$ and $L_{23} = 1.5\lambda$. Thus, knowing ϕ, we get the exact δd needed for estimating θ in Equation (4).

Disambiguating $\boldsymbol{\theta}$ Using Equation (4) and the techniques discussed thus far, we can determine $\sin\theta$ between the compass and a particular beacon B. But as Figure 6 shows, in general, there are two locations B_1, B_2 for a beacon B that result in the same θ at the compass. This is due to symmetry of the system about the line X–X. An analytical way of understanding this is to observe that there are two values of θ in the range $[0, 2\pi)$ for a given value of $\sin\theta$. This ambiguity in the location of the beacon prevents us from determining a unique value for the heading.

We solve this by using *two* sets of non-collinear receiver-triplets to break the symmetry. We place the two sets of receiver-triplets perpendicular to each other as shown in Figure 7, and there can be now be only one position for the beacon B. We are now given an angle θ_1 relative to X-X and θ_2 relative to Y-Y, which means that $\sin\theta_1$ and $\sin\theta_2$ are known. It is easy to see that there can only be a unique solution for this configuration. These two perpendicular sets of receiver-triplets are configured using *five* receivers on the compass, as shown in Figure 8.

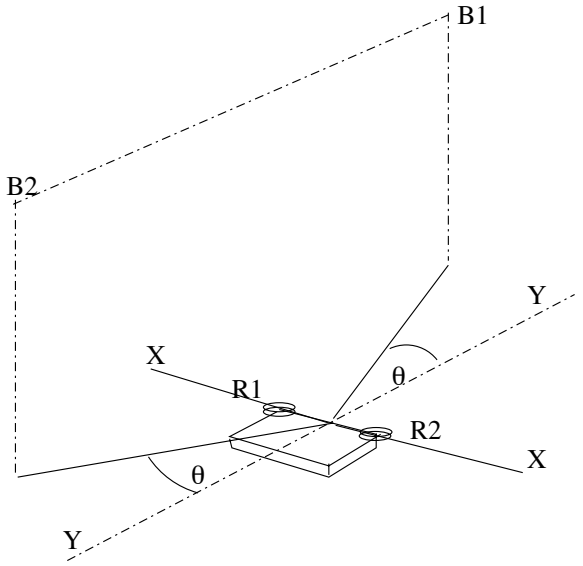

Fig. 6. θ is ambiguous—the beacon can be at either $B1$ or $B2$.

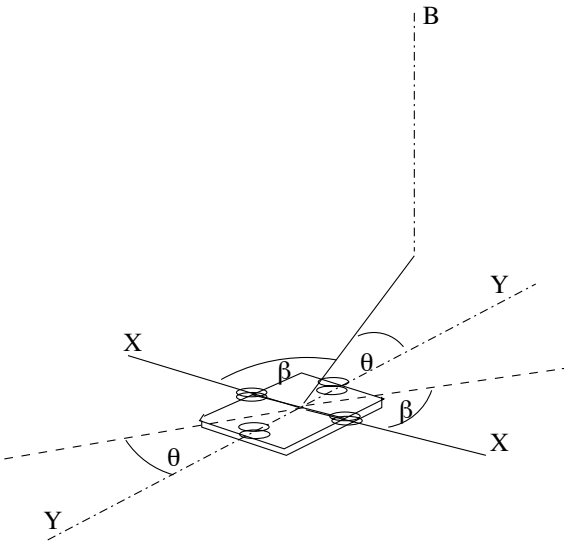

Fig. 7. Two sets of receivers can break the symmetry. One set of receiver triplets lies on the X-X line and the second set lies on the Y-Y line.

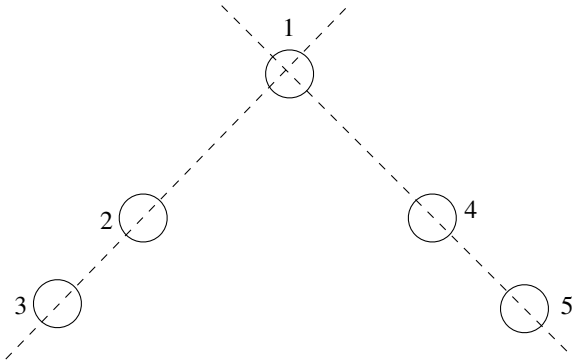

Fig. 8. Five receivers on a compass forming two perpendicular receiver-triplets, which are used to unambiguously infer the heading with respect to a beacon.

2.3 Determining compass coordinates

In the previous discussion we assumed that the receiver can determine the ratio of height to distance, $z/\bar{d}$, to a particular beacon. To enable the receiver to gather this information, we implement a coordinate system using a number of active beacons instrumented with known positions within the space. The compass determines its mean position as an (x,y,z) tuple by listening to beacon transmissions. This mechanism also enables us to determine the speed of sound in the vicinity of the compass.

Both the Bat and the Cricket systems use a combination of RF and ultrasound signals to measure distances, using the relative speeds between these two signals. However, to determine the distance accurately, it is necessary to know the speeds of both signals. The speed of RF is essentially infinite in our setting, but the speed of ultrasound depends on environmental factors such as temperature and humidity. The Bat system compensates for this variation by measuring environmental factors. The Cricket system is robust against such variation by virtue of its dependence only on relative distances.

We present a technique that enables us to determine the position in terms of (x,y,z) coordinates using 4 beacons without knowledge of the speed of sound or requiring additional environmental sensors. We use the measured propagation time $\hat{t_i}$ to each beacon, which is *proportional* to the actual distance d_i.

We implement a coordinate system within the room assuming the ceiling to be the x-y plane and z to be positive inside the room (down-

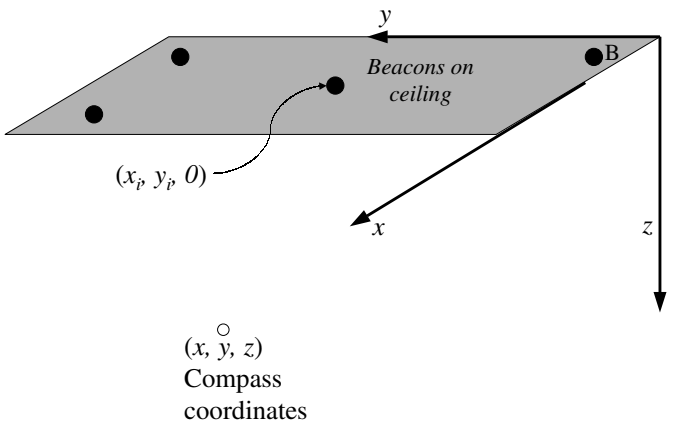

Fig. 9. The coordinate system used in Cricket; the beacons are configured with their coordinates and disseminate this information on the RF channel.

wards), as shown in Figure 9. Consider four beacons B_0, B_1, B_2, and B_3 attached to the ceiling of a room. Each beacon B_i broadcasts its coordinates $(x_i, y_i, 0)$ on the RF channel, which is sensed by the receivers on the compass. At the same time, it also broadcasts an ultrasonic pulse. A receiver, which is at an unknown coordinate (x,y,z), measures the time difference $\hat{t}_i$ between the arrival of these two signals to beacon B_i. The actual distance from the receiver to B_i is therefore equal to $v\hat{t}_i$, where v is the (unknown) speed of sound.

We can then write the following family of four equations for the unknowns (x, y, z, v). Recall that we are interested in the value of $z/\bar{d} = z/\sqrt{x^2+y^2+z^2}$.

$$(x - x_i)^2 + (y - y_i)^2 + z^2 = v^2\hat{t}_i^2, \;\; 0 \leq i \leq 3 \tag{10}$$

We can eliminate z^2 from these equations by subtracting each equation from the previous one, to obtain the following three *linear* equations in the three variables, x, y and v^2:

$$A \times \begin{bmatrix} x \\ y \\ v^2 \end{bmatrix} = \begin{bmatrix} x_1^2 - x_0^2 + y_1^2 - y_0^2 \\ x_2^2 - x_1^2 + y_2^2 - y_1^2 \\ x_3^2 - x_2^2 + y_3^2 - y_2^2 \end{bmatrix} \tag{11}$$

where the matrix A is given by

$$A = \begin{bmatrix} 2(x_1 - x_0)\ 2(y_1 - y_0)\ (\hat{t}_1^2 - \hat{t}_0^2) \\ 2(x_2 - x_1)\ 2(y_2 - y_1)\ (\hat{t}_2^2 - \hat{t}_1^2) \\ 2(x_3 - x_2)\ 2(y_3 - y_2)\ (\hat{t}_3^2 - \hat{t}_2^2) \end{bmatrix}$$

If the determinant of A is non-zero, then Equation (11) can be solved to determine unique values for x, y, and v^2. Substituting these values into Equation (10) then yields a value for z^2, whose positive square root yields z. Furthermore, we can use this estimate of v, to further improve the accuracy of the $d_2 - d_1$ estimation by using a better estimate of the wavelength of the ultrasonic carrier.

However, the ability to determine x, y, z, and v^2 uniquely from above set of equations depends on the values (x_i, y_i) for $0 \leq i \leq 3$. Specifically, one can show that if the beacons B_0,B_1,B_2, and B_3 are placed such that they do not all lie on the same straight line or circle, then there is always a unique solution to the above set of equations. In practice, this placement can be done easily by placing 4 beacons on the corners of a rectangle and then by moving one of the beacons some distance along the diagonal of the rectangle. An example placement of the beacons is shown in Figure 9.

Superficially, the equations above are similar to those used by GPS receivers to determine receiver position. In GPS, the beacons are satellites with precise clocks; latency from satellite to receiver is non-negligible; the propagation velocity is known (to first order) as the speed of light; and a system of equations is solved at the receiver to recover the receiver's absolute position and time [15]. In contrast, in Cricket, propagation time (for RF) is negligible; the beacons have no clocks; and the propagation speed (for ultrasound) is unknown. Our system also solves for four unknowns, three of position and one for the speed of sound in the local medium.

3 Implementation

We have implemented prototypes of the beacon and compass hardware described in Section 2. Each beacon is configured with its position in a coordinate system, which it broadcasts on a 418 MHz RF channel. Concurrent with each periodic RF broadcast, it sends a 500 μs ultrasonic pulse at 40 KHz, which are received at the compass ultrasonic receivers. Each beacon and compass has an on-board PIC microcontroller that implements the communication protocol and processes information. The rest of this section describes the details of the communication protocol between the beacons and compass, and how the

compass processes the observed differential distance estimates to deduce the actual differential distances.

3.1 Protocol details

The beacons in Cricket operate in an autonomous manner, without any centralized control of when they transmit information [20]. To reduce inter-beacon interference at the receivers, each beacon senses the RF carrier before transmitting a locally unique ID and its known position coordinates. In addition, each subsequent transmission is sent at a uniformly chosen random time after the previous one. In our implementation, the average amount of time between successive transmissions is 250 ms. The packet format of the beacon includes information about the geographic space (e.g., an intentional name for resource discovery in INS [1], a URL as in CoolTown [8], etc.). Each packet is protected using a block-parity code. The compass detects collision on the RF channel and discards samples that do not pass a block-parity check, which helps it disambiguate between potentially interleaved RF/ultrasound combinations sent of separate beacons.

The processing of ultrasonic signals is more involved. The compass hardware does analog-to-digital oversampling to detect low-to-high transitions from each ultrasonic receiver. In addition to processing RF information, the on-board PIC microcontroller handles the ultrasonic signals received by the several ultrasonic receivers on the compass to obtain phase difference estimates, and passes these to the software running on the attached device.

This software processes the raw data to obtain observed differential distance estimates, and then convert them to actual differential distance estimates. It also infers the coordinates of the compass relative to the coordinate system defined by the beacons, and computes the orientation unit vector in that system. It calculates the angle relative to each beacon and uses the smallest angle to derive the orientation vector. The reason for this will be clear from Section 4, which shows that the accuracy of our system worsens at large angles (greater than about 45 degrees). This also means that the system works best when it finds at least one beacon at an angle smaller than 45 degrees—since there are at least four beacons per space on each ceiling of interest, it is relatively straightforward to place, and find, at least one beacon in standard rectangular rooms.

3.2 Differential distance estimation algorithm

In our prototype, the ultrasonic receivers are set up according to Figure 8, where $L_{12} = L_{14} = 2\lambda$, and $L_{23} = L_{45} = 1.5\lambda$. An interesting aspect of our implementation is the method used to determine the unique actual differential distance from the observed differential distance. The method uses the intuition developed in Section 2.2, where an "existence" argument was made for how to configure receivers to unambiguously resolve the actual phase difference. Although the argument was made in the "phase domain," the results hold equivalently in the "wavelength domain," where the measured values are the differential distances in terms of λ. However, the argument in Section 2.2 is not prescriptive, so we outline our implemented algorithm below.

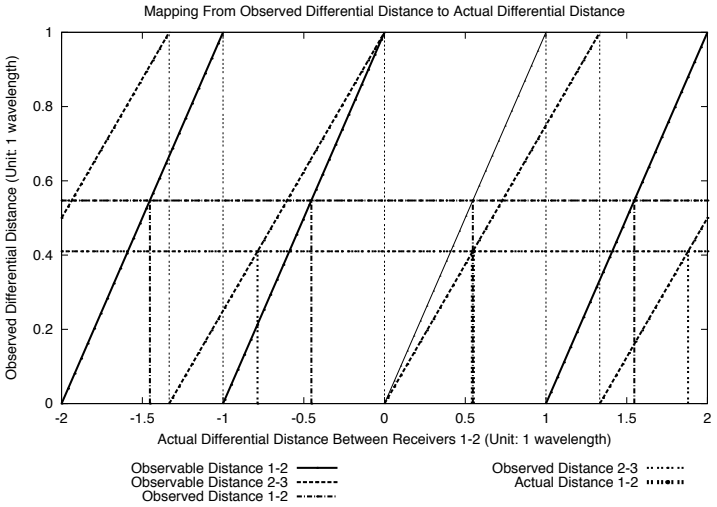

Fig. 10. Finding the actual differential distance between R_1 and R_2 by using the observed differential distances from (R_1, R_2) and (R_2, R_3).

Consider Figure 10, which plots the variation of observed differential difference $\delta d'$ as a function of the actual differential difference δd for two pairs of receivers. One of the curves (the solid line segments) shows the $\delta d'_{12}$ variation for the receiver pair (R_1, R_2), which are separated by a distance $L_{12} = 2\lambda$. The other curve (the dashed line segments) shows the variation $\delta d'_{23}$ for the receiver pair (R_2, R_3) separated by $L_{23} = 1.5\lambda$. We normalized the curves to show the observed variations

of $\delta d'_{12}$ and $\delta d'_{23}$ as a function of δd_{12}; i.e., $\delta d'_{12}$ varies in the range $[0, \lambda]$ as δd_{12} varies in $[-2\lambda, 2\lambda]$.

Each curve is periodic with discontinuities. The observed value $\delta d'$ varies in the range $[0, \lambda]$ because that is the range of measurable distance between two (time-shifted) waveforms whose starting times are not known. The discontinuities are due to the fact that the observable differential distances follow the periodicity of the observed phase differences. The actual differential distances vary in the range $[-L_{12}, L_{12}]$ for δd_{12}, and in the range $[-L_{23}, L_{23}]$ for δd_{23}. But because we have normalized the curves as a function of δd_{12}, the observed phase differential curve for the receiver pair (R_2, R_3) shown in Figure 10 also varies in the range $[-L_{23} \cdot L_{12}/L_{23}, L_{23} \cdot L_{12}/L_{23}] = [-L_{12}, L_{12}]$ in the plot. The slope of each line segment is proportional to the normalized separation distance for that pair of receivers. Hence, the normalized curve for (R_1, R_2) has a slope of 1, while the curve for (R_2, R_3) has a slope of $L_{23}/L_{12} = 3/4$.

Note that because L_{12} and L_{23} are relatively prime multiples of $\lambda/2$, the periods (and discontinuities) for the two curves always differ, and the cycle of each curve (i.e., the discontinuities) do not overlap each other more than once. Consequently, the two curves do not have a repeating pattern within the interested range $[-L_{12}, L_{12}]$. Hence, we get a unique solution for the actual δd value for *any* given pair of observed $\delta d'_{12}$ and $\delta d'_{23}$ values.

Recall that the range of observable differential distances is $[0, \lambda]$. From Figure 10, we see that any observed value within this range can be mapped to four possible solutions for the actual δd_{12}. Let $A^{\delta d'_{12}}$ be the set of possible solutions derived from the observed value $\delta d'_{12}$. Graphically, these are the values on the horizontal axis extrapolated from the four intersections between the $y = \delta d'_{12}$ line and the observable differential distance curve for the receiver (R_1, R_2). Then, given an observed $\delta d'_{12}$, our task is to identify the actual differential distance from the set $A^{\delta d'_{12}}$.

By following the arguments presented in Section 2.2, we can use the observed $\delta d'_{23}$ to help us identify the correct solution. From Figure 10, the observed $\delta d'_{23}$ can be mapped to three possible solutions for the actual δd_{12}. Again, let $A^{\delta d'_{23}}$ be the set of possible solutions using the observed value $\delta d'_{23}$. Since we are guarenteed a unique solution for any given pair of observed values $\delta d'_{12}$ and $\delta d'_{23}$, we will find exactly one matching solution that exists in both $A^{\delta d'_{12}}$ and $A^{\delta d'_{23}}$. Thus, the final answer for the actual differential distance δd_{12} is a if and only if $a \in A^{\delta d'_{12}}$ and $a \in A^{\delta d'_{23}}$.

For example, Figure 10 shows that for the observed $\delta d'_{12} = 0.547$ and $\delta d'_{23} = 0.41025$, $A^{\delta d'_{12}} = \{-1.453, -0.453, 0.547, 1.547\}$ and $A^{\delta d'_{23}} = \{-0.786, 0.547, 1.880\}$. Hence, the final solution is $\delta d_{12} = 0.547$ because this value exists in both $A^{\delta d'_{12}}$ and $A^{\delta d'_{23}}$.

One caveat about this algorithm for finding the actual phase differential distance is that measurement errors may produce no matching solution that exists in both $A^{\delta d'_{12}}$ and an $A^{\delta d'_{23}}$. In such a situation, we find the closest matching solution by choosing an $a_{12} \in A^{\delta d'_{12}}$ and $a_{23} \in A^{\delta d'_{23}}$ such that $|a_{12} - a_{23}|$ is minimum. Then, we report the actual differential distance to be $\frac{a_{12}+a_{23}}{2}$.

4 Experiments

In this section, we report on several performance experiments conducted with our Cricket compass implementation. In Section 5, we outline a few improvements that we intend to implement in the future, based on what we have learned from these experiments.

We describe two distinct sets of experiments. First, we evaluate the efficacy of our differential distance estimation technique as a function of the angle θ between the compass and one fixed beacon using the techniques of Section 2.2. Then, we attach multiple beacons at different places on a ceiling and measure the accuracy of coordinate estimation using the techniques of Section 2.3. Finally, we combine the results of these experiments to perform an analytic error analysis of Equation (4) to derive an upper bound on the end-to-end errors one might expect in practice. We do this because our current prototype hardware does not allow us to obtain the average and differential beacon distances simultaneously; while we are building this combined hardware, we do want to get a sense of how accurate our system is likely to be. The following sections demonstrate that our differential distance and position estimation methods work well.

4.1 Differential Distance Estimation

In this set of experiments, we use the setup shown in Figure 2 to measure the accuracy of the differential distance values, $d_2 - d_1$, at different values of θ. We place the beacon such that it is at a height $z = 1.5m$, a horizontal distance of $x = 2.0m$ away from the receivers, and an angle θ with respect to the line joining the beacon and the receiver. The receivers are configured according to Figure 5, where $L_{12} = 2\lambda$, $L_{23} = 1.5\lambda$.

For each measurement at the specified θ, we take the *mode* of the differential distance samples to reduce the error caused by ultrasound reflections and noise. The entire experiment was repeated for three trials.

The results are shown in Figure 11, which shows the average angle estimates derived from the measured $d_2 - d_1$ values. That is, the average angle estimates were calculated by applying Equation (4) on the known values of z, x, and the average measured $d_2 - d_1$ values at each θ. From Figure 11, we see that the measured $d_2 - d_1$ values can estimate the true angle with reasonable accuracy.

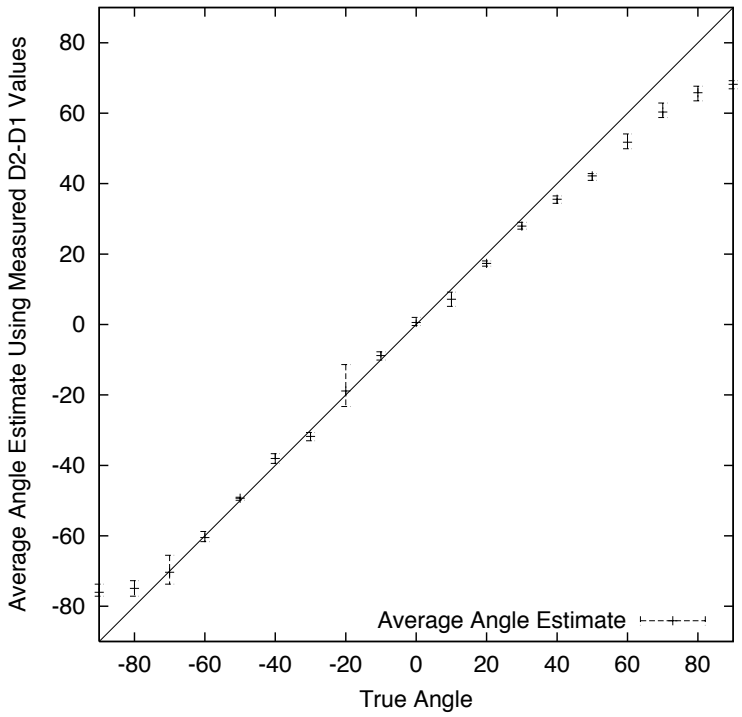

Fig. 11. Average angle estimates versus true angle values. The error bars indicate the absolute angle errors for all three trials. The line $y = x$ plots the ideal relationship between the true and estimated angle values.

Table 1 shows the average differential distance in terms of λ, the average percentage error of the differential distance for every 10° angle $-90° \leq \theta \leq 90°$, and the derived angle estimates.

$\theta°$	Actual $d_2 - d_1$ (λ)	Measured $d_2 - d_1$ (λ)	Error of $d_2 - d_1$(%)	Derived θ Estimates ($\theta°$)
-90	-1.600	-1.552	3.000	-76.021
-80	-1.576	-1.544	2.011	-74.901
-70	-1.504	-1.504	-0.033	-70.350
-60	-1.386	-1.392	-0.459	-60.482
-50	-1.226	-1.213	1.007	-49.319
-40	-1.028	-0.985	4.193	-38.021
-30	-0.800	-0.843	-5.333	-31.786
-20	-0.547	-0.516	5.707	-18.894
-10	-0.278	-0.245	11.699	-8.821
0	0	0.021	—	0.57
10	0.278	0.200	-28.015	7.184
20	0.547	0.477	-12.773	17.359
30	0.800	0.749	-6.333	27.929
40	1.028	0.931	-9.509	35.573
50	1.226	1.075	-12.320	42.202
60	1.386	1.256	-9.356	51.755
70	1.504	1.389	-7.594	60.317
80	1.576	1.459	-7.427	65.794
90	1.600	1.485	-7.167	68.196

Table 1. Differential distances (measurements averaged over 3 trials), percentage error and the derived angle estimates at each value of θ

We make three observations from these results. First, we are able to accurately estimate angles to within ±3 degrees in the range from -70 to 30 degrees, and to within ±8 degrees for angles up to 50 degrees. Second, in terms of percentage error[2] all estimated differential distances (and hence, angles for this set of experiments) have less than 13% error with the exception of $\theta = 10°$. Third, the estimates of the positive θ values consistently show a higher percentage error than those for negative θ. Moreover, they all report a value that is less than the true value. Our current hypothesis, still under active investigation, is that the causes of these errors are imperfect calibrations of the distances

[2] For angle estimation, the percentage error is not as interesting to most applications as absolute errors.

between the ultrasonic receivers (which a better mounting process will fix) and timing delay issues related to interrupt handlers in the PIC microprocessor on the compass. Despite these caveats, we find the ability to estimate angles to within 3-to-5 degrees for a practical range of angles promising and useful for many context-aware applications.

4.2 Distance and Position Estimation

Beacon	(x, y, z)
H	0, 121, 0
I	117, 121, 0
J	0, 0, 0
K	117, 0, 0

Table 2. Beacon coordinates (in centimeters).

For our second set of experiments, we placed four beacons on the ceiling of a room at known coordinates as shown in Table 2. Each beacon broadcasts a unique identifier, which is mapped to its known coordinates by the receiver. The receiver is placed at a specific location and collects up to 25 distinct distance samples from each of the four beacons. Because the noise and reflections of ultrasound in the environment can affect the sampled distances, we take the mode of each distance distribution for each beacon as the actual distance estimate to each beacon. We configured the compass as in the previous experiment, taking measurements at four different compass locations as shown in Table 3. At each location, we collected data across four independent trials.

Table 3 shows the coordinate estimates at each locations. We find that our position estimates can be accurate to within 5-6 centimeters, and in the worst case, to within 25 centimeters of the true value. We conducted a further investigation into the worst case situation—although it was good to less than a foot, we were interested in the underlying reasons for this behavior. We found that consistently the worst case happened when the receiver was near a wall, or when the beacon was attached close to a wall, while it was possible to obtain centimeter-level accuracy a few feet away from walls. We have since been developing techniques to handle reflections from walls, which we outline in Section 5.

Receiver Location	Actual Receiver Location (x, y, z)	Estimated Receiver Location (x, y, z)	Error (cm)
A	0.0, 121.0, 178.0	-13.8, 134.9, 193.8	25.12
B	117.0, 121.0, 178.0	123.3, 129.1, 190.3	16.05
C	0.0, 0.0, 178.0	-0.7, -5.5, 176.8	5.65
D	117.0, 0.0, 178.0	120.4, -3.0, 173.1	6.63

Table 3. Coordinate estimates at four different receiver locations.

From the coordinate estimates, we also derive a set of $z/\bar{d}$ values that are used in Equation (4). Table 4 reports the percentage error of the $z/\bar{d}$ derived from our coordinate estimates. *The results indicate that the error in $z/\bar{d}$ is at most* 2.6%*, even when near a wall, and substantially better further away.* We use this worst-case measured data in the next section to understand the theoretical error bound on overall orientation estimation using our compass.

Receiver Location	Percentage Error of z/d with Respect to Beacon			
	H	I	J	K
A	-0.50	-0.98	-0.92	0.03
B	0.38	-0.15	0.38	1.26
C	-1.65	-1.57	-0.05	-0.58
D	-2.59	-1.71	-1.75	0.28

Table 4. Percentage error of $z/\bar{d}$ with respect to each beacon for each coordinate estimation trial.

4.3 Error Analysis

We now give a simple error analysis of the angle estimation method, obtaining an expression for how it depends on the errors in the measured quantities. We use our experimental results from the previous sections that bound the accuracy with which our techniques and implementation estimate $(d2 - d1)$ and $z/\bar{d}$.

If $V(v_1, v_2)$ is a function of two *independently-measured* variables v_1 and v_2, then the error in V, ΔV, can be expressed as [6, 23]:

$$(\Delta V)^2 = (\frac{\partial V}{\partial v_1})^2(\Delta v_1)^2 + (\frac{\partial V}{\partial v_2})^2(\Delta v_2)^2 \tag{12}$$

Applying this to the angle estimate θ, we get the following expression for the fractional error in θ, $\Delta\theta$, as a function of $v_1 = d_2 - d_1$ and $v_2 = z/\bar{d}$:

$$\frac{\Delta\theta}{\theta} = \frac{\tan\theta}{\theta} \times \sqrt{(\frac{\Delta v_1}{v_1})^2 + (\frac{\Delta v_2}{v_2})^2 \frac{v_2^4}{(1 - v_2^2)}} \tag{13}$$

Note that because $d_2 - d_1$ is estimated using the phase difference of the ultrasonic waveforms, and $z/\bar{d}$ is estimated using a different method combining the RF and ultrasonic signal arrival times, v_1 and v_2 satisfy the independent-measurement considerations of Equation (12).

Equation (13) shows that the error might grow to be rather large, especially for values of θ close to $\pi/2$. The physical reason for this is apparent from Figure 3, which shows that at large values of θ, small changes in x2 produce large changes in θ. Equation (13) also shows that the error might grow large when the $z/\bar{d}$ value is small. The physical reason is clear from Figure 2, which shows that as the receiver moves closer to the beacon, a small change (or error) in the differential distance produces a large change in θ.

We now apply Equation (13) to the average error values from our experimental measurements to obtain an upper bound on the expected error. For $x = 2.0m$ and $z = 1.5m$, we get $v_2 = z/\bar{d} = 0.6$. We then set $\frac{\Delta v_2}{v_2} = 0.0259$, which is the worse average case error for $z/\bar{d}$ from Table 4. We then substitute the θ and $\frac{\Delta v_1}{v_1}$ values from Table 1. The projected theoretical upper error bound at each θ is listed in Table 5.

We find that the theoretical upper bound on error is less than five degrees when θ is between ± 40 degrees. We emphasize that this is what the theory predicts as an upper bound for each θ, and that in practice things may well be better (and are in fact better in some cases, as our reported experiments showed).

θ	$\tan\theta/\theta$	dv_1/v_1 (%)	$\Delta\theta/\theta$ (%)	$\Delta\theta^\circ$
-90	∞	3.00	∞	∞
-80	4.06	2.01	9.44	7.55
-70	2.25	0.03	2.62	1.84
-60	1.65	0.46	2.07	1.24
-50	1.37	1.01	2.10	1.05
-40	1.20	4.19	5.23	2.09
-30	1.10	5.33	6.02	1.81
-20	1.04	5.71	6.07	1.21
-10	1.01	11.70	11.88	1.19
0	1.00	—	—	—
10	1.01	28.02	28.33	2.83
20	1.04	12.77	13.37	2.67
30	1.10	6.33	7.10	2.13
40	1.20	9.51	11.51	4.61
50	1.37	12.32	16.90	8.45
60	1.65	9.36	15.59	9.36
70	2.25	7.59	17.28	12.09
80	4.06	7.43	30.54	24.43
90	∞	7.17	∞	∞

Table 5. Projected error bounds for angle estimations at each θ. The parameters are $v_2 = 0.6$ and $\frac{\Delta v_2}{v_2} = 0.0259$.

4.4 Effect of Motion

The experiments mentioned above were conducted by placing the compass on a stable platform (i.e., the linear velocity of the compass is zero). In practice, we expect the Cricket compass to be attached to mobile devices, and are interested in measuring its performance when a user walks or moves the device in their hand. We model such movement as a linear velocity and calculate the Doppler effect to examine the performance impact of such movement on the Cricket compass.

Let λ' be the observed wavelength of ultrasound due to motion, λ be the true wavelength of ultrasound from the beacon, f be the true frequency of ultrasound, and v_r be the linear velocity of the receivers in the direction towards the beacon. Then, because of the Doppler effect, we get:

$$\lambda' = \lambda - \Delta\lambda = \lambda - v_r/f$$

We use Equation (5) to derive the error $\Delta(\delta d)$ caused by the Doppler effect: z

$$\Delta(\delta d) = \frac{\phi}{2\pi} \cdot \Delta\lambda \;\; ; \;\; \Delta\lambda = \frac{v_r}{v_s} \cdot \lambda$$

where v_s is the velocity of sound. Hence, at a pedestrian walking speed of $v_r = 2.0m/s$ and $v_s = 330m/s$, $\Delta\lambda = 0.006\lambda$. In our implementation of the software compass, $\phi = 0.556\pi$ at $\theta = 10°$, so the error with respect to the true δd is about 1.2%. At $\theta = 40°$, the error is less than 1%.

5 Improvements

The preliminary experiments reported in the previous section show great promise, and we believe that this augurs well for the utility of our system. However, our results also raise some important issues that need to be addressed in implementation before a production system can be realized. This section describes some of these issues and our proposals to address them.

5.1 Handling Reflections

Four appropriately placed beacons can accurately estimate the position coordinates of a receiver, but our results show that the accuracy degrades when a beacon is within a few inches from a wall. This is because ultrasound reflections can cause the measured distances to be inaccurate. If there is a line-of-sight path between the beacon and the receiver, we will have a single correct[3] distance among the set of distances; if not, then several of the readings will be incorrect.

We can solve the ambiguity caused by multiple distances and errors due to incorrect distances by using *five* beacons instead of four. With five beacons, the receiver will have a set of readings containing multiple measured distances to each beacon. Now, from this set, the receiver can select four beacon values at a time, each value corresponding to a different beacon, and run the algorithm of Section 2.3 to determine its coordinate position. If the coordinates determined from two or more distinct sets of beacons are close to each other, we can select that as the correct coordinate. Otherwise, we cannot have much confidence in the correctness of the estimated coordinates (although they will likely be correct to a few inches).

[3] Here "correct" refers to a distance that is proportional to the actual distance.

Here, we essentially use the fifth beacon to validate the coordinates obtained using the other four; the robustness of this scheme is based on the assumption that the probability of two incorrect readings $\hat{d}_1$ and $\hat{d}_2$ giving rise to answers that coincide is negligible. An analogy might help understand why this is reasonable: Consider a line segment of length l joining two points, P_1 and P_2. We are told that a point in between them is at d_1 and d_2 away from P_1 and P_2 respectively. If both $\hat{d}_1$ and $\hat{d}_2$ are independent (and incorrect) estimates of d_1 and d_2, it is highly unlikely that the errored values will correspond to the same identical point!

5.2 Handling Diffractions

Another potential cause of error is the diffraction (bending at edges) of sound waves around obstacles. Such obstacles may not block the entire path but cause the signal to bend. If the signal arriving at the receiver is bent, then the measure angle to the beacon will have a corresponding error. The difference in distance due to bending could be on the order of millimeters, which will not be detected by the method described above since the error in the distance would be the same order of magnitude (or even less) than the accuracy of distance measurement itself.

However, the receiver can determine its orientation with respect to a fixed origin using each of the beacons it can hear from, and use values that coincide to be the right one. We intend to modify our current method of using the smallest angle and replace it with this "plurality" scheme.

5.3 Beacon and Compass Placement

One of the issues that a production-style deployment of the compass infrastructure must pay close attention to is beacon placement. From Equation 12, it is clear that the error is large when θ is large, and also when z is close to $\bar{d}$. What we would like is to ensure that, for every compass location, there is at least one beacon whose θ from that location is smaller than 45 degrees. In addition, we would also like to ensure that there is at least one visible beacon whose $z/\bar{d}$ is not bigger than some threshold value, say 0.5. This second condition means that there should be at least one beacon whose height does not "dominate" the distance to the compass, i.e., the compass should not be "directly under" all visible beacons.

For most rectangular rooms, these conditions are rather straightforward to meet without requiring a large number of beacons. In general,

however, a more formal approach will be valuable to tackle this placement problem using ideas from the classical "art-gallery" problems and more recent "searchlight" problems in computational geometry. To our knowledge, constraints similar to our compass system have not been studied in the literature, and this area is open to interesting algorithm development, especially for non-rectangular rooms.

Some of the discussion in this paper assumes (perhaps tacitly) that the compass is held flat and parallel to the ground. This is not a fundamental requirement—with this requirement, all we need is the orientation with respect to one beacon, while otherwise we need the orientation with respect to at least three beacons to uniquely determine the orientation vector. Since we have at least four beacons for coordinate determination, this is not hard to accomplish.

6 The Viewfinder

We have developed the *Viewfinder* application to demonstrate the use of the location and orientation information provided by the software compass. The user defines a sweep angle β and a distance R and points a device running the Viewfinder in the desired direction. The Viewfinder then highlights the services discovered within the swept sector.

To enable this functionality, the Viewfinder application queries a resource discovery server, such as those proposed in [1, 9, 13], to obtain the global coordinates of the available services. To facilitate the bootstrapping process, the name of the server for the space is advertised on the RF channel by the beacons. We also assume that individual services use their own software compass to obtain their coordinate information, and that they advertise this information to the resource discovery system. Otherwise, a system administrator can assign global coordinates to each individual (static) service.

The Viewfinder queries the software compass for current values of the relative angle θ with respect to the beacon B, the coordinates of B, and the coordinates of the device's current location O. Then, to test whether a service S is within the user-specified sweep angle, the Viewfinder extends two vectors originating from the device's coordinates: one to B and one to S. From these vectors, the Viewfinder invokes the cosine law to find a unique solution $\theta_S = \angle SOB$, which is the angle of the service S with respect to the anchor beacon. Then the Viewfinder simply performs a series of comparisons between the relative angle values θ_S, θ, and ϕ_S to test whether S lies within the current sweep angle, and at a distance smaller than the user-specified distance.

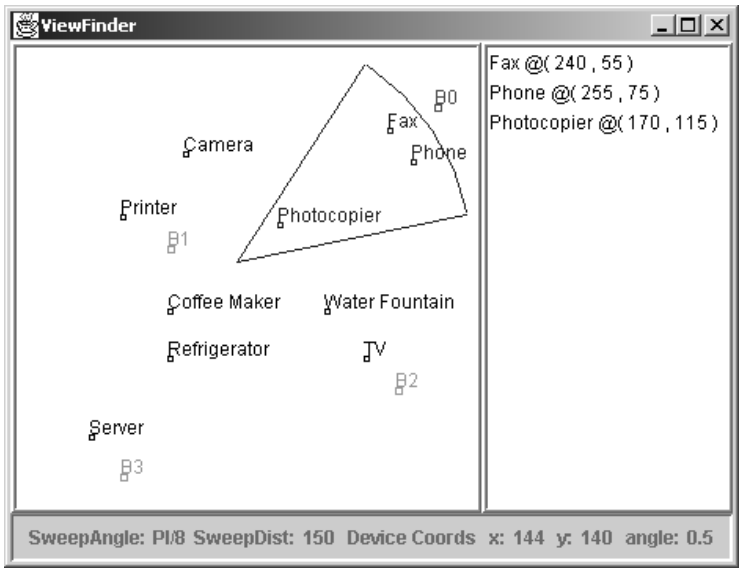

Fig. 12. Screen capture of the Viewfinder application. Note that the origin is at the upper left corner of the map and the angle is reported in radians, where the angle value starts at zero due east and increases counterclockwise.

Figure 12 shows a screen capture of the prototype Viewfinder application. A map of the service and beacon locations is displayed on the left panel. The tip and the body of the pie-shape figure marks the device location and its current sweep angle. Services that are within the current sweep angle appear on the right panel. The bottom panel displays the coordinate and angle values reported by the software compass.

7 Related Work

Want *et al.*'s Active Badge system, developed using infrared links, was one of the earliest indoor systems for position inference [26]. Its architecture inspired future generations including the Bat system [27] and PinPoint's local positioning system [18, 29]. In these architectures, the hardware tag attached to mobile devices is active, and responds to queries from a central controller and location database about its whereabouts. While the Bat system uses a combination of RF and ultrasound to estimate distance [14, 27], PinPoint uses spread-spectrum radio signals and multiple antennae at the controller to process messages from a tag. One of the problems with these architectures is that they track users, and lead to significant privacy concerns [19].

Bahl and Padmanabhan describe RADAR [5], an indoor RF-based location system that uses an already-existing data network to estimate position. Here, the RF signal strength is used as a measure of distance between RF transmitter and a receiver. This information is then used to locate a user using triangulation, typically using an RF signal strength map obtained by a prior instrumentation process.

Our compass system enhances the capabilities of the Cricket location system, which uses a combination of passive receivers (called "listeners") and active beacons, which provide information about a space [20]. Like the Bat system, it uses a combination of RF and ultrasound to estimate position, but uses multiple ultrasonic receivers located close to each other to infer orientation on a mobile handheld device.

The best-known system for outdoor use is the satellite-based Global Positioning System (GPS) [12, 15], which is increasingly being used in civilian applications in addition to its traditional military use. GPS does not provide the degree of precision required for mobile applications indoors because of the low RF signal strength, high RF noise, and the reflections of RF signals due to the presence of metallic objects. Bulusu *et al.* describe a low-cost location system for outdoor use [7], where the environment is instrumented with a number of fixed RF stations that periodically transmit their unique ID and position. The receivers use

RF connectivity to estimate their position relative to the known fixed RF stations.

Doherty *et al.* model the position estimation problem in ad hoc sensor networks as a convex optimization problem, showing that under some conditions it is possible for the nodes to discover their positions relative to one another [10]. Savvides *et al.* describe another approach to this problem that resembles our coordinate estimation scheme of Section 2.3. We expect variants of these approaches to be a good starting point for instrumenting beacons in our environment without having to program each beacon with its location, but programming only some of them and having the others discover their coordinates from the other beacons in their vicinity.

The Constellation system uses a combination of accelerometers, gyros, and ultrasonic sensors to estimate position and orientation [11]. Like Cricket, the Constellation relies on an active set of ultrasonic beacons to determine the initial tracking position of the device and then recursively refines the orientation estimation using information gathered by the inertial sensors. However, the tight coordination that is required between the receivers and transmitters of this system makes it unsuitable for large-scale indoor deployment. It is also unclear that this can be implemented in a handheld-like form factor.

The HiBall system uses opto-electronic tracking of hundreds or thousands of infra-red LEDs mounted in special ceiling panels [28]. It provides rapid updates of receiver position and orientation, but requires the installation of large arrays of LEDs in the ceiling and carefully machined camera at the client, which will significantly increase deployment costs.

Commerical magnetic motion trackers have been used in virtual reality and simulation applications such as head-mounted displays and biomechanic motion capture: Ascension [2], Startrak [24], and Aurora [3] are three products available today. They provide reasonably accurate estimates of the position and orientation of the target object by sending magnetic pulses and detecting the change of field strength along three orthogonal axes. These systems usually requires a centralized coordination between the magnetic transmitters and receivers and are susceptible to magnetic interference from the presence of metals or other conductive materials in the environment [16], which causes problems in many indoor environments.

Roumeliotis *et al.* describe the implementation of an orientation sensor that uses a Kalman filter to combine a compass and robot odometry with a absolute orientation signal from a "sun sensor" [21]. This system

works under kinematic conditions, and its approach may be combined with Cricket to improve our system.

8 Conclusion

The Cricket compass system described in this paper reports position and orientation indoors, for a handheld, mobile device, and informs an application running on the device of the position and orientation in a local coordinate system established by the fixed set of beacons. To our knowledge, this is the first handheld-integrated system that provides a combination of orientation and position information to within a few degrees of the true value indoors, making it an attractive technology for various context-aware pervasive computing applications. It does not require large or fast user motions and works even when a traditional magnetic compass fails. The hardware configuration consists of a microcontroller, one RF receiver, and five ultrasonic receivers placed in a "V" shape a few centimeters across, processing 418 MHz RF data and 40 KHz ultrasonic signals sent from active beacons.

The challenges in deriving orientation for a small device arise from the need for sub-centimeter differential distance estimates, and from the need for accurate position estimation. We solved the first problem using multiple carefully placed receivers, deriving the mathematical conditions for placement. We solved the second problem by developing a position estimation technique that compensates for the unknown velocity of sound in an environment by observing propagation times and explicitly calculating it. Our experimental results show that we can obtain angles to within about 3 degrees of the true value in most practical settings.

9 Acknowledgments

We thank Franklin Reynolds for useful discussions on the motivation for the Cricket compass. We thank David Andersen, Rui Fan, Stephen Garland, Lars-Ake Larzon, and Suchitra Raman for several helpful comments and suggestions on previous versions of this paper.

This work was funded by NTT Inc. under the NTT-MIT research collaboration, by Acer Inc., Delta Electronics Inc., HP Corp., NTT Inc., Nokia Research Center, and Philips Research under the MIT Project Oxygen partnership, and by IBM Corp. under a university faculty award.

References

1. Adjie-Winoto, W., Schwartz, E. and Balakrishnan, H. and Lilley, J. The design and implementation of an intentional naming system. In *Proc. ACM Symposium on Operating Systems Principles*, pages 186–201, Kiawah Island, SC, December 1999.
2. Ascension Technology. `http://www.ascension-tech.com/`, 2000.
3. Northern Digital Inc. - ProductsAURORA. `http://www.ndigital.com/aurora.html`, 2001.
4. R. Azuma. Tracking requirements for augmented reality. *Comm. of the ACM*, (7):50–55, July 1993.
5. P. Bahl and V. Padmanabhan. RADAR: An In-Building RF-based User Location and Tracking System. In *Proc. IEEE INFOCOM*, Tel-Aviv, Israel, March 2000.
6. Y. Beers. *Introduction to the Theory of Error*. Addison-Wesley, Reading, MA, 1957.
7. N. Bulusu, J. Heidemann, and D. Estrin. GPS-less Low Cost Outdoor Localization For Very Small Devices. Technical Report 00-729, Computer Science Department, University of Southern California, April 2000.
8. CoolTown. `http://www.cooltown.hp.com/`, 2000.
9. S. Czerwinski, B. Zhao, T. Hodes, A. Joseph, and R. Katz. An Architecture for a Secure Service Discovery Service. In *Proc. 5th ACM MOBICOM Conf.*, pages 24–35, Seattle, WA, August 1999.
10. L. Doherty, K. Pister, and L. Ghaoui. Convex position estimation in wireless sensor networks. In *Proc. IEEE INFOCOM*, April 2001.
11. E. Foxlin, M. Harrington, and G. Pfeiffer. Constellation: A Wide-Range Wireless Motion-Tracking System for Augmented Reality and Virtual Set Applications. In *Proc. ACM SIGGRAPH*, Orlando, FL, July 1998.
12. I. Getting. The Global Positioning System. *IEEE Spectrum*, 30(12):36–47, December 1993.
13. Y. Goland, T. Cai, P. Leach, Y. Gu, and S. Albright. Simple Service Discovery Protocol/1.0. `http://search.ietf.org/internet-drafts/draft-cai-ssdp-v1-02.txt`, June 1999. Internet Draft, expires December 1999.
14. A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster. The Anatomy of a Context-Aware Application. In *Proc. 5th ACM MOBICOM Conf.*, Seattle, WA, August 1999.
15. B. Hoffmann-Wellenhof, H. Lichtenegger, and J. Collins. *Global Positioning System: Theory and Practice, Fourth Edition*. Springer-Verlag, 1997.
16. V. Kindratenko. Calibration of Electromagnetic Tracking Devices. *Virtual Reality: Research, Development, and Applications*, 4:139–150, 1999.
17. Oxygen home page. `http://oxygen.lcs.mit.edu/`.
18. Pinpoint home page. `http://www.pinpointco.com/`.
19. Privacy international survey. `http://www.privacyinternational.org/survey/technologies.html`.

20. N. Priyantha, A. Chakraborty, and H. Balakrishnan. The Cricket Location-Support System. In *Proc. 6th ACM MOBICOM Conf.*, Boston, MA, August 2000.
21. S. Roumeliotis, G. Sukhatme, and G. Bekey. Smoother-based 3-d attitude estimation for mobile robot localization. In *Proc. IEEE International Conf. on Robotics and Automation*, Detroit, MI, May 1998.
22. B. Schilit and M. Theimer. Disseminating Active Map Information to Mobile Hosts. *IEEE Network*, pages 22–32, Sep/Oct 1994.
23. B. Shchigolev. *Mathematical Analysis of Observations*. Iliffe Books Ltd., London, 1965. Originally published in the U.S.S.R. in 1960 (Russian).
24. Polhemus Star Trak. `http://www.polhemus.com/stardstech.htm`, 2000.
25. T. Turunen, T. Pyssysalo, and T. Lankila. Utilisation of Wireless Application Protocol to Implement Mobile Augmented Reality Based Services. In *Proc. W3C and WAP Workshop on Position Dependent Information Services*, February 2000. Available from `http://www.w3.org/Mobile/posdep/Oulu.html`.
26. R. Want, A. Hopper, V. Falcao, and J. Gibbons. The Active Badge Location System. *ACM Transactions on Information Systems*, 10(1):91–102, January 1992.
27. A. Ward, A. Jones, and A. Hopper. A New Location Technique for the Active Office. *IEEE Personal Comm.*, 4(5):42–47, October 1997.
28. Greg Welch, Gary Bishop, Leandra Vicci, Stephen Brumback, Kurtis Keller, and D'nardo Colucci. The HiBall tracker: High-performance wide-area tracking for virtual and augmented environments. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, December 1999.
29. J. Werb and C. Lanzl. Designing a positioning system for finding things and people indoors. *IEEE Spectrum*, 35(9):71–78, September 1998.